Fast Adaptive Multipole Method for Computation of Electrostatic Energy in Simulations of Polyelectrolyte DNA

MARCIA O. FENLEY and WILMA K. OLSON*

Department of Chemistry, Rutgers, the State University of New Jersey, New Brunswick, New Jersey 08903

KIAT CHUA and ALEXANDER H. BOSCHITSCH

Continuum Dynamics, Inc., P.O. Box 3073, Princeton, New Jersey 08543

Received 21 March 1995; accepted 3 August 1995

ABSTRACT

This article presents a fast adaptive method for the computation of long-range electrostatic interactions in computer simulations of polyelectrolyte DNA. Classically, the computation of electrostatic energy involves a direct summation of all pairwise interactions due to the charged phosphate groups in the molecule. This results in an N-body interaction problem with an asymptotic time complexity of $O(N^2)$ which is computationally very expensive and limits the number of phosphate groups that can be used in computer simulations of polyelectrolyte DNA to at most several hundred. We describe an effort to speed up computer simulations of polyelectrolyte DNA with the use of a fast adaptive hierarchical algorithm for the computation of electrostatic energy (i.e., modified Debye–Hückel energy). The asymptotic time complexity is reduced to $O(N)$ with the implementation of the fast hierarchical algorithm on serial computers. This is achieved by grouping phosphate groups into an adaptive hierarchical data structure and computing the interactions between groups using low order multipole and Taylor series expansions expressed in Cartesian coordinates. We first examine the accuracy and speed enhancements of the fast hierarchical method in the computation of the electrostatic energy of circular DNA at zero and high salt concentrations. The fast hierarchical method is further tested in a one-step Monte Carlo (MC) simulated annealing algorithm for closed circular supercoiled DNA. In all cases, we observe order of magnitude reductions in the computation time with negligible loss of numerical accuracy in the electrostatic energy computation.

* Author to whom all correspondence should be addressed.

Journal of Computational Chemistry, Vol. 17, No. 8, 976–991 (1996)

CCC 0192-8651 / 96 / 080976-16

Introduction

DNA POLYELECTROLYTE

As a polyelectrolyte, double-helical DNA is profoundly affected by ionic interactions. Positive counterions neutralize the negatively charged phosphate groups along its two complementary sugar-phosphate backbones and impart electrostatic stability to the system. The nature of this effect, "counterion condensation," is unique to polyelectrolytes and involves many competing factors including polymer chain configuration, ionic concentration, and ion type (size and shape).[1] Computer simulations of the electrostatic properties of DNA and its complexes with drug and proteins are emerging, nevertheless, as an important counterpart to experimental work.[2] Modeling and structural analysis of nucleic acids, however, is not yet as routine as that of either proteins or drugs.[3-5] Explicit treatment of DNA, water, counterions, and bound ligands at concentrations of biological interest and/or experimental accessibility are still beyond the capabilities of even the most sophisticated computers.[6] The size-limitation problems confronting computer simulations of nucleic acids necessitate the use of primitive models where polyions and ions are treated explicitly and where solvent water is represented by a dielectric continuum.[7] In addition, when chains of interest are longer than those accessible to detailed all-atom treatments, the DNA must often be simplified to an approximate atomic model (consisting of one charged phosphate group per nucleotide residue). Such considerations motivate the present interest in fast algorithms for this problem.

For superhelical DNA, salt effects are particularly important since many parts of the double helix that are distant in the linear sequence come into close contact upon supercoiling. The typical energy model of supercoiled DNA is a simple elastic energy function with bending and twisting components appropriate for an uncharged, isotropic, long thin rod with circular cross section.[8] Long-range interactions are commonly treated with a hard-sphere excluded volume potential that only crudely mimics the electrostatic repulsions of contacted segments.[9-11] Clearly, more appropriate potentials that monitor the long-range nature of electrostatic interactions are essential for correct prediction of the structure, properties, and interactions of DNA. Enumeration of the nonbonded interactions presents a formidable task, but one that can be addressed with the fast adaptive multipole method presented here.

Until recently, computational studies of charged supercoiled DNA molecules have been based on highly simplified molecular representations of uncharged linearly elastic polymers. Fenley et al.[12] recently incorporated the electrostatic and elastic behavior of DNA together with excluded volume effects in MC/simulated annealing studies of small (100–200 base pairs) closed circular superhelical DNAs. In these simulations, the polyelectrolyte character of the charged phosphate groups of DNA was accounted for by including a modified Debye–Hückel (i.e., electrostatic) energy term. To date, computer hardware limitations have precluded: the study of longer DNA chains at natural levels of supercoiling, the comparison of different electrostatic energy functions, the detailed examination of specific physiological salt concentrations, and the full treatment of the many competing factors governing the polyelectrolyte character of DNA. These issues must be examined to account properly for experimental effects of salt on the structure and energetics of supercoiled DNA.[13-20]

N-BODY PROBLEM

As is well known, the electrostatic (e.g., Coulombic or Debye–Hückel) potential term involves long-range interactions between all charged particles in a system and is classically evaluated by direct summation of pairwise contributions (hereafter referred to as the direct method). Given N particles, the computation of the electrostatic energy has an asymptotic time complexity of $O(N^2)$. This severely constrains the size of problems that can be solved using the direct method to at most several hundred charged particles since reasonable computing times cannot be obtained for much larger N. Attempts at reducing the number of operations through the use of standard cutoff distance criteria are confounded by the slow decay of the electrostatic potential, which requires that all pairwise charge–charge interactions be accounted for. Several studies have demonstrated that standard cutoff schemes can introduce significant errors in forces, energies, and other observable quantities in molecular simulations.[21-26] Clearly, methods that alleviate the computational burden of the direct method without forfeiting accuracy are needed to realize routine electrostatic calculations using realistic detailed models con-

taining orders of magnitude more charged particles than are currently feasible.

The computational difficulties of the direct method are further multiplied in molecular dynamics and MC simulations of polyelectrolyte DNA. Typical simulations proceed over a very large number of steps, each involving an electrostatic energy calculation. One can see that such N-body simulations are not feasible even with projected near term supercomputer capabilities. Consequently, simulations of polyelectrolyte DNA have been limited until now to short DNA supercoils, and rigorous theoretical studies of electrostatic effects of realistic superhelical DNA (i.e., by means of large-scale computer simulations) have not been possible. Any significant improvement in the efficiency of the simulations of polyelectrolyte DNA can be achieved only by a speedup of the electrostatic energy computation.

The computational intensiveness of the electrostatic interaction problem encountered in molecular simulations of DNA structure is a feature of the classical N-body problem faced by researchers in a wide range of scientific and engineering disciplines, including chemistry/biochemistry,[27-45] potential theory,[46-48] fluid dynamics,[49-55] combustion,[56-58] materials science,[59-62] astrophysics,[63-66] and plasma physics.[67,68] The review articles by Greengard[69,70] delineate several fields where the N-body problem is encountered.

FAST COMPUTATIONAL METHODS

In the literature, numerous methods have been put forth to address the computational complexity of N-body problems.[52,55,64,65,71-79] The most effective of these methods are the $O(N)$ and $O(N \log N)$ hierarchical schemes that have been proposed in recent years.[64,65,77,78] In such methods, particles are grouped into a hierarchy of clusters whose dimensions span the entire spectrum of length scales. Each characteristic length scale is thus identified with a hierarchical level, and the relative group sizes or levels and their separations are used to distinguish between far-field and near-field interactions. In the former, the interaction between entire, widely separated groups can be approximated using a far-field approximation. For closely located near-field groups, such approximations do not apply and the interactions between individual particles contained in the groups must be treated directly.

The various fast hierarchical methods differ in the treatment of far-field interactions and the generation of hierarchical groups. Appel[64] first developed a hierarchical method for astrophysical N-body simulations. His version relies on a single term monopole approximation of the gravitational potential and employs a binary-tree data structure whose leaves or terminal nodes represent individual particles and whose branching nodes represent clusters of particles. Another hierarchical fast method which retains up to quadrupole order terms in the approximation to the gravitational potential was subsequently introduced by Barnes and Hut.[65] Their scheme involves a hierarchical subdivision of physical space into regular cells (an oct-tree in three dimensions). Both algorithms have $O(N \log N)$ computational time complexity. The reader is referred to the original articles for more details concerning these methods.[64,65]

Greengard[77] and Carrier et al.[78] generalized these earlier developments by formulating an arbitrary order fast multipole method (FMM) for electrostatic calculations. In addition to constructing a formal framework for arbitrary order expansions, they also obtained rigorous error bounds for the approximations, thus making systematic accuracy control possible to within a specified tolerance. The method is based on the use of complex power series and spherical harmonic expansions in two and three dimensions, respectively. The number of terms retained in the expansions is chosen to achieve a prescribed level of accuracy. The computational complexity of the FMM is proportional to the number of particles [i.e., $O(N)$]. Recently, White and Head-Gordon[79] presented new developments in the formulation, implementation, and performance of the FMM for electrostatic potential and force calculations. The error estimates due to key FMM approximations and the linear scaling of the FMM with increasing number of particles were studied in detail.

Hierarchical methods that allow particle–group interactions (e.g., the Barnes and Hut, BH, algorithm) generally give rise to $O(N \log N)$ behavior, whereas those that also include group–group interactions (e.g., the FMM) are more complex but can achieved $O(N)$ time complexity. It is important to recognize, however, that the distinction between $O(N)$ or $O(N \log N)$ asymptotic time complexity is usually of minor importance since significant differences in computation time are only realized for values of N considerably larger than those contemplated in DNA simulations. The leading constant is likely to be of much greater significance with regard to actual computing time, and this constant will be influenced by the particular

hierarchical scheme used (e.g., Appel, Barnes, and Hut; Greengard and Rokhlin, GR), the extent of the near-field, and the number of terms employed in the expansions. For example, the 2^k-pole multipole GR algorithm requires $875k^4N$ operations including the expensive evaluation of high-order spherical harmonics. Hence, despite the $O(N)$ asymptotic behavior of the method, the leading constant is quite large so that it can render the scheme costly for moderate error bounds and/or N. An important area warranting additional research is to establish optimal combinations of expansion order and near-field extents (i.e., regions wherein direct summation rather than multipole expansion is employed) that minimize computation time for a given number of particles and desired accuracy.

The original version of the FMM is based on a nonadaptive clustering scheme where every first level (coarsest) cell is subdivided an equal number of times and the number of cells grows exponentially with the refinement level. Such an approach is best suited for a reasonably uniform distribution of particles. On the other hand, an adaptive form of the FMM such as the one developed by Carrier et al.[78] is better equipped to treat nonuniform particle distributions. In the adaptive scheme, only cells with sufficiently many particles are further subdivided so that cell refinement is directly related to the local particle density. The selection of an adaptive versus a nonadaptive scheme implies major differences in the overall code structure since in the former case one must allow for, and distinguish between, interactions involving differently sized groups.

MACROMOLECULAR APPLICATIONS

Many approaches for calculating long-range nonbonded interactions in energy minimization and molecular dynamics simulations of macromolecules have been recently implemented by several groups.[26-32,39-45,59,61,62] Saito[30] developed the particle–particle and particle–cell method (PPPC), which is based on the BH algorithm, to represent the Coulomb potential properly and to allow simulation of proteins in solution without cutoffs. No use of a Taylor series expansion is made in his work, the efficiency gain being entirely due to the multipole expansions invoked for far-field interactions.

A nonadaptive fast multipole[77] algorithm combining both multipole and Taylor series expansions was adopted by Shimada et al.[43] for biomacromolecular simulations. This method formulates expansions in terms of Cartesian coordinates instead of spherical harmonics. Ding et al.[59] developed the cell multipole method (CMM) for computing long-range nonbonded interactions (including London dispersion and shielded Coulombic terms) in energy minimization and molecular dynamics simulations of polymeric systems. This implementation of the FMM is also based on a Cartesian formulation of the expansions and uses low-order (only up to third-order) expansions as well as a nonadaptive clustering scheme. The CMM has been further employed in molecular dynamics simulations of very large proteins, such as the protein capsid of tomato bushy stunt virus.[39] Board et al.[31,41,42] formulated a fast hierarchical multipole algorithm for computation of Coulombic and London interactions in molecular dynamics simulations of large proteins in water. Their method is based upon high-order multipole and Taylor series expansions and employs a nonadaptive multiple-level clustering scheme. The multipole series expansion of the Coulombic and Lennard–Jones ($1/r^6$ term) potentials is based on spherical harmonics and ultraspherical harmonics frameworks, respectively.

A structure adapted multipole method (SAMM) was developed by Niedermeier and Tavan[44] for studying protein dynamics. This formulation does not employ local Taylor expansions and considers terms in the multipoles only up to second order. In contrast to all the above methods, SAMM employs a hierarchical decomposition scheme adapted to a set of predefined structural units of the protein rather than chosen rigidly as a cubic grid.

OVERVIEW

In this article we describe a fast adaptive multipole method for the computation of the electrostatic (i.e., modified Debye–Hückel) energy of polyelectrolyte DNA. In our method, low-order multipole and Taylor series expansions are rooted in a Cartesian approach, using regular spatial moments as the basic building blocks, since this allows more efficient evaluation and shifting of multipole coefficients. Both particle–group and group–group interactions are retained, making $O(N)$ behavior possible. An adaptive clustering scheme is adopted due to our interest in highly nonuniform particle distributions. The scheme also embodies the first attempt at incorporating Debye–Hückel screening effects into the framework of multipole/Taylor series expansions.

We first apply the fast adaptive multipole method to compute the electrostatic energy of a simple DNA circle under the conditions of no added salt and high monovalent salt concentration (i.e., 1 M NaCl). The fast adaptive multipole method is further employed in one step computer simulations of polyelectrolyte DNA at both salt conditions mentioned. A comparison of the fast method results with those obtained with the direct method is made along with an assessment of the computational speed enhancement, robustness, and numerical accuracy of the fast adaptive multipole method. In the next sections we describe our DNA representation, energy model, and the fast method employed in electrostatic energy computations of polyelectrolyte DNA.

DNA Chain Representation

Our theoretical study of supercoiled DNA employs a simplified chain model instead of an elaborate all-atom representation since we are interested in the global macroscopic properties of the double helix. The axis of the closed circular duplex is represented by a series of piecewise cubic *B*-spline curves. Each curve segment is defined by a polynomial expression which weights the relative contributions of a given set of controlling points.[80] The *B*-spline construction guarantees chain closure and geometric continuity and can be used to describe complex trajectories of the double-helical axis with a small number of variables.[9, 10, 12, 81]

To account for the polyelectrolyte nature of DNA, the negatively charged phosphate groups are placed at evenly spaced increments of 1.7 Å along the chain contour. For simplicity, the structure of each phosphate group is reduced to a single point centered at the phosphorus atom. This spacing corresponds to the projection of phosphate groups of complementary strands on the helical axis of the canonical B-DNA duplex.[82] The coordinates of the phosphorus atoms are generated using an approximation algorithm of computational complexity $O(N)$ as outlined below.

To obtain the coordinates, the composite cubic *B*-spline curve is divided into equidistant segments and rescaled so that the arc length between neighboring phosphorus atoms is nearly constant. The first step in atom positioning consists of generating points s_0, s_1, up to s_m, where s_m coincides with s_0 and m is much greater than N, the total number of phosphate groups of the DNA. Consecutive points on the curve are unevenly spaced but much closer together than the desired spacing. The curve is approximated by a series of straight line segments between adjacent points and the total contour length (l) of the curve computed from:

$$l = \sum_{j=1}^{m} |S_j - S_{j-1}|, \tag{1}$$

where S_j and S_{j-1} are the respective coordinates of points s_j and s_{j-1} in the *B*-spline curve.

To insert N equally spaced points ($t_0, t_1, \ldots, t_N$ where t_N coincides with t_0) on the curve, the computed contour length of the curve is divided by N and t_0 set to s_0. The points t_i are then chosen inductively such that the distance between t_i and t_{i-1} is $\sim l/N$. That is, $k(i)$ (where $k(i) \in \{0, m\}$) is chosen inductively so that,

$$\left| \left(\sum_{j=k(i-1)+1}^{k(i)} |S_j - S_{j-1}| \right) - \frac{l}{N} \right| < c, \tag{2}$$

and then let $t_i = s_{k(i)}$. In this work, $c = 10^{-5}$ and $m \approx 100N$ to ensure that the distance between t_i and t_{i-1} is very close to l/N. The values of m and c determine the accuracy of the algorithm. Finally, the coordinates of the equally spaced points are multiplied by the factor $(1.7 \times N)/l$ so that the distance between adjacent phosphorus atoms is ~ 1.7 Å.

Energy Computation

DIRECT METHOD

To account for the polyelectrolyte nature of DNA (due to the negatively charged phosphate groups in the sugar-phosphate backbone) in numerical simulations of DNA, the modified Debye–Hückel potential given by:

$$(q')^2 \sum_{k<j}^{N} \frac{e^{-\kappa r}_{kj}}{\varepsilon r_{kj}}, \tag{3}$$

is employed where N is the total number of phosphate groups, r_{kj} is the distance between the kth and jth phosphate groups, ε is the dielectric constant of water at 25° (i.e., 78.3), and q' is the effective phosphate charge, reduced by $(1 - N_v\theta_{N_v})$ to account for counterion condensation.[1] Since for B-DNA there is 76% neutralization of the phosphate charges by bound monovalent (Na^+) ions [i.e., $(1 - N_v\theta_{N_v}) = 0.24$], the effective charge q' is

given by $0.24q$ where q is the unit electrical charge. The parameter θ_{N_v} is the number of condensed N_v-valent counterions per DNA charge and N_v is the valence of the salt (e.g., $N_v = 1$ for NaCl). The Debye screening parameter κ is defined by:

$$\kappa = 0.329\sqrt{c_s}, \quad (4)$$

where c_s is the concentration of NaCl in molarity units. Two extreme cases will be considered: no added salt ($c_s = 0$ M) and high salt concentration ($c_s = 1$ M).

Direct summation of all $N \times (N-1)/2$ pairwise electrostatic interactions in eq. (3) is computationally intensive, effectively limiting values of N to several hundred and thus rendering large-scale simulations of DNA molecules using the direct method impossible. In the following subsections we describe a fast adaptive multipole method designed to reduce the computational load associated with the electrostatic energy computation.

FAST METHOD

Far-Field Approximation

Central to the successful reduction in the time complexity of the fast algorithm is the decomposition of the electrostatic energy computation into a near-field treatment and a far-field approximation. This drastically reduces the CPU effort expended in calculating the interactions between distant particles while ensuring that a high level of accuracy is preserved. The essential procedure is as follows. A three-dimensional grid of hierarchically nested cubic cells is first adaptively generated about the particle distribution and each particle is identified with its enclosing box. These cells thus represent groups or clusters of particles whose interactions can be classified as near- or far-field, depending upon the group separation distance. Interactions between particles residing within the same cell or immediate neighboring cells are considered near-field and are evaluated directly using eq. (3). All other particles are considered far-field entities and their interactions may be approximated using multipole and Taylor series expansions for the entire group.

A schematic of the far-field group–group interaction is shown in Figure 1. Given two groups A and B, the effect of group A acting upon group B, is computed by first developing a multipole representation about the center of group A. The "induced" effect (electrostatic potential and its spatial gradients) is then evaluated at the center of group B, and subsequently extrapolated to individual particle locations within group B using a Taylor series expansion. Below we describe how the multipole and Taylor series expansions can be used in the computation of long-range energy interactions of the form $1/r^n$ where r is the distance between the source and observation points and n is an arbitrary positive integer.

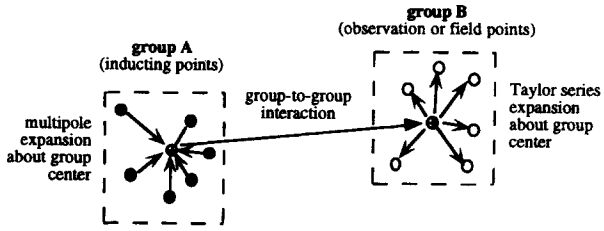

FIGURE 1. Schematic showing the far-field group–group interactions between particles (i.e., charged phosphate groups) on a DNA chain.

Multipole and Taylor Series Expansions

To describe the key concepts involved in the fast method, consider the case where the Debye screening parameter, κ, is set to zero. Then, eq. (3) can be written as:

$$U_E = \frac{1}{2\varepsilon}\sum_{j=1}^{N} q_j u_j, \quad u_j \equiv \sum_{i \neq j}^{N} \frac{q_i}{r_{ij}}, \quad (5a,b)$$

where U_E is the electrostatic energy of a system of N charged particles of strength, q_i, located at R_i and the separation distance, $r_{ij} = |R_i - R_j|$. The fast algorithm is applied to each of the N terms u_j, which are specific evaluations of the general function:

$$u(R) = \sum_{i \neq j}^{N} \frac{q_I}{|R - R_i|^n}, \quad (6)$$

at $R = R_j$ with $n = 1$. Assuming that the source points, R_i, belong to a spatially localized group and that the observation point, R, is well separated from this source group, then eq. (6) can be represented by a multipole expansion. Selecting as origin any convenient point located within the

source group, the multipole expansion of arbitrary order is given by:

$$u(R) = \frac{1}{r^n}\sum_{k=0}^{\infty} E_k \frac{1}{r^k}, \quad E_k = \sum_{i=1}^{N} C_k^{n/2}(\hat{e}\cdot\hat{e}_i) r_i^k q_i, \tag{7a,b}$$

where E_k are the moment coefficients of the expansion, $r = |R|$, $r_i = |R_i|$, $\hat{e} = R/r$, and $\hat{e}_i = R_i/r_i$. The coefficients $C_k^{n/2}$ are formally given by the Gegenbauer polynomials:

$$C_k^{\lambda}(t) \equiv \left(-\frac{1}{2}\right)^k \frac{\Gamma(2\lambda + k)}{\Gamma(2\lambda)} \frac{\Gamma\left(\lambda + \frac{1}{2}\right)}{\Gamma\left(\lambda + \frac{1}{2} + k\right)} \times \frac{(1-t^2)^{(1/2)-\lambda}}{k!}\frac{d^k}{dt^k}(1-t^2)^{\lambda+k-(1/2)}, \tag{7c}$$

where $\lambda = n/2$ and Γ is the gamma function. The use of the Gegenbauer polynomials facilitates the expansion to arbitrary high order and allows methodical derivation of the expansion for different powers, n. For the electrostatic calculations considered here, $n = 1$ and eq. (7a) can be written in the form:

$$u(R) = \frac{1}{r}\left\{\left(\sum_i q_i\right) + \frac{1}{r}\hat{e}\cdot\left(\sum_i q_i D_i\right) + \frac{1}{2r^2}\hat{e}\cdot\left(\sum_i q_i [Q]_i\right)\cdot\hat{e} + \cdots\right\}, \tag{8}$$

where $D_i = R_i$ is a vector, $[Q]_i = 3R_iR_i^T - r_i^2[I]$ is a second order tensor, and $[I]$ is the identity matrix of order three. In the above equation, the terms in parentheses on the right-hand side are the monopole, dipole, and quadrupole, respectively. These summations are independent of the observation point, R, and need only be calculated *once* for each group in preparation for the fast calculation. Therein lies the key computational advantage in expressing the far-field interactions in the form of eq. (8)—the calculation is separated into determining a set of multipoles for each group and then evaluating group–group interactions by a limited set of inner products, thus resulting in significant computational savings.

Provided that the observation point R is well separated from the group of charged particles, then convergence of the multipole expansion is assured. In particular, the truncation error, e, of the multipole expansion is bounded by:

$$e < C(d/r)^{p+1}, \tag{9}$$

where p is the number of terms in the expansion, r is the distance between the groups, d is the diameter of the group containing the charged particles, and C is a constant.[77]

Further CPU reduction is realized by combining the multiple expansion with a Taylor series approximation at the observation points. Thus, assuming that M observation points are clustered in physical space (i.e., group B in Fig. 1), then eq. (8) need only be evaluated at the group center, R_c, of the observation *group*. The induced effects at each of the individual observation points are subsequently obtained by a Taylor series expansion about this center:

$$u(R) = u(R_c) + \delta R \cdot \left(\frac{\partial u}{\partial R}\right)\Big|_{R_c} + \frac{1}{2}\delta R \cdot \left[\frac{\partial^2 u}{\partial R^2}\right]\Big|_{R_c}\cdot \delta R + \cdots, \tag{10}$$

where $\delta R = R - R_c$. In eq. (10), the gradients are evaluated only once at R_c and the electrostatic potential u at each observation point is obtained through simple multiplication. This is computationally less costly than evaluating eq. (8) (which involves the expensive square root) for every observation point, particularly when the number of observation points is large. The gradient terms can be obtained directly by differentiating the expansion, eq. (8), or else by analytical differentiation of eq. (6) and expanding the resulting expression using the Gegenbauer polynomials.

In the preceding development it was assumed that $\kappa = 0$. For nonzero κ, the following approach is adopted. Near-field interactions are evaluated directly by scaling the energies between individual particle pairs by $e^{-\kappa r}$ where r is the distance between the *particles*. For the far-field interactions, the proper way to proceed is to develop a formal multipole expansion of the term $(e^{-\kappa r})/r$. There are then three characteristic distances involved: the *group* separation distance (r), the characteristic group size (d), and the Debye length ($1/\kappa$). Up to five distinct far-field ranges (each having $d \ll r$) can be considered: $\kappa d \ll 1$, $\kappa r \ll 1$; $\kappa r \sim 1$ (hence $\kappa d \ll 1$); $d \ll 1/\kappa \ll r$; $\kappa d \sim 1$ (hence, $\kappa r \gg 1$); and $1 \ll \kappa d \ll \kappa r$. One can show that for the first and third cases the far-field interactions can be approx-

imated by the Taylor series expansion of $(e^{-\kappa r})/r$. For the fourth and fifth cases the exponential decay dominates the algebraic behavior, and one finds that the errors associated with simply multiplying the $\kappa = 0$ multipole expansion by $e^{-\kappa r}$ are consistent with those already incurred when truncating the $\kappa = 0$ expansion. In the present study a simpler approach is adopted. The induced effect of a source group upon a field group is first calculated using the multipole expansion for $\kappa = 0$. This effect is then scaled by $e^{-\kappa r}$ (r now being the distance between *groups*). This approximation is formally valid for the fourth and fifth ranges above. For cases one to three, the approximation is correct to the first-order term only. Nevertheless, numerical tests comparing the approximate and direct electrostatic energy predictions of circular DNA arrangements for κ ranging from 0 to 1 indicate that errors remain acceptable (i.e., of comparable magnitude to the $\kappa = 0$ case), including the intermediate range ($\kappa = 0.01$ to 0.1) where errors are expected to be largest. We anticipate incorporating the formal multipole expansion of $(e^{-\kappa r})/r$ in the future.

Adaptive Hierarchical Scheme

In a single-level scheme, the domain is subdivided into equal size cubical boxes and near- and far-field interactions are easily distinguished in terms of box indices. The time complexity of the fast algorithm on a single level grid is $O(N^{4/3})$.[54] To reduce this further to $O(N)$ or $O(N \log N)$, a hierarchical grouping technique is adopted where multiple levels of nested cells are generated (see Fig. 2). The first level of cells consists of a set of relatively large cubes covering the domain of interest. Each first-level cell is divided into eight second-level cells which in turn are divided further into smaller third-level cells, etc. This process continues until every cell either contains no more than a specified number of particles, or has achieved the maximum allowable refinement level. A hierarchy of "meshes" is thus produced and used to define the interacting groups or clusters and to distinguish between near- and far-field interactions. The decision to employ multipole and/or Taylor series expansions between any two cells is formulated entirely in terms of their separation distance and cell dimensions.

This hierarchical structure acts as a sieve for grouping the interactions between the computational points. Let the levels be numbered, $l = 0, L$ with L the finest level. At the first level, $l = 0$, the

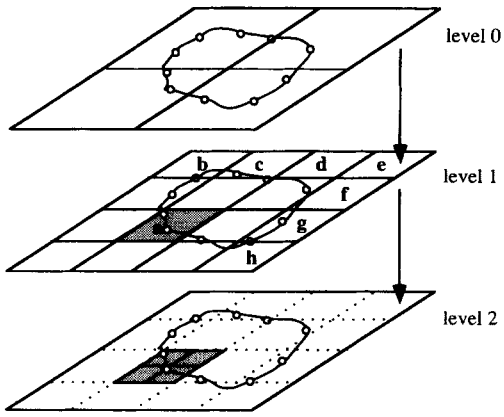

FIGURE 2. Schematic showing a hierarchical data structure with splitting of a cell into subcells at lower levels.

interactions between the well-separated cells are computed and the remaining interactions between elements which are in the same cell or immediate neighboring cells are ignored because they will be considered in a subsequent level. The level 0 interactions are passed down to level 1 using the Taylor series expansion, eq. (10), and the level 1 interactions between all nonneighboring subcells (e.g., the interactions between subcell a and subcells b–h in Fig. 2) are carried out. The combined induced effects of level 0 and 1 interactions are in turn passed down to level 2 where the above process is repeated. At the finest level, exact point–point (near-field) interactions are also computed. This constitutes the "pass down" or downward traversal of the hierarchical data structure since interactions are accumulated down to the finest level.

Multipole coefficients for all levels are required and the process just described is preceded by an accompanying upward traversal of the hierarchy where coarser level multipole coefficients are constructed from their finer level constituents. Thus, at the finest level $l = L$, the moment coefficients are computed directly by summing the individual particle contributions about the cell centers. At the next coarser level, $l = L - 1$, the moment coefficients of a cell (e.g., the level 1 cell, a, in Fig. 2) are constructed from their L-level counterparts using algebraic shifting rules which can be efficiently implemented because of the simple nested cell topology. The $(L - 2)$ level coefficients can in turn be deduced from the $(L - 1)$ quantities in like manner and the entire process continued to the coarsest level, $l = 0$. The bulk of the calculation

lies in constructing the finest level moments, with coarser level entities being readily obtained at only a fraction ($< 1/7$ in three dimensions) of the cost required for the finest level construction.

To realize the advantages of a multilevel strategy fully, the hierarchical algorithm is implemented using an adaptive approach where cell subdivision only occurs in regions containing particles rather than over the entire domain. For inhomogeneous particle distributions (e.g., DNA chains), an adaptive algorithm dramatically reduces the total number of boxes and hence storage compared to uniform subdivision of the original coarse level grid where the number of boxes, $N_b \sim 8^L$. Since most of these boxes are empty, it is clearly wasteful to store them. The key issue regarding an adaptive versus a nonadaptive grid is the memory requirement, because the CPU time is essentially the same for both methods, provided one does not run into memory constraints (A. H. Boschitsch, unpublished results).

To generate the adaptive hierarchical tree one begins with the coarse first-level grid and inserts the particles in succession. When the number of particles contained in a given box exceeds a user-specified threshold, that box is subdivided into eight subcells and the original particles are reassigned to these child boxes. Concurrently, indexing arrays specifying the box refinement level, parent box index, and child box index are developed in conjunction with a pointer array specifying the particles contained by a leaf box (i.e., one that has not been subdivided). The reverse relationship specifying the enclosing leaf box for every particle is also established. These relationships are all expressed in terms of compact pointer arrays to minimize memory cost.

Determining Interaction Lists

An adaptive strategy is more difficult to implement than one where boxes are refined uniformly since there is no longer a simple correspondence between a box index and its location. Additional tests are necessary to prevent double counting (i.e., ensure that the interaction between a pair of boxes is not already implicitly contained in a previous coarser level interaction). Furthermore, since interactions may involve boxes of different size, only one of the multipole or Taylor series expansions may apply. Hence, the most difficult aspect of the adaptive fast implementation is to determine for each box, *ib*, the set of interacting boxes, *jb*, and the approximations permitted in computing the effect induced by box *jb* upon box *ib*.

Procedurally, the fast algorithm is implemented by looping over the observation (or field) boxes. For each box, the contributing source boxes and the manner in which they interact with the field box are established. These interactions are then computed and, if the box has children, the induced parameters and their spatial gradients are projected onto the descendent box centers using Taylor series expansion. Finally, if the box is a leaf, induced effects are extrapolated to each of its resident particles and the remaining near-field interactions are carried out.

To establish the types of interactions between any pair of boxes, the following terms related to the hierarchical data structure are first defined:

Descendant A box, d, is a descendant of another box, a, if d lies within the region covered by a. In tree-structure terminology, box a is traversed in the path from d to the root node.

Ancestor If box d is a descendant of box a, then a is the ancestor of d.

Child The child, c, of box, p, is one of its immediate descendants, i.e., the level of a child box is one greater than that of p.

Parent If box, c, is a child of p, then p is denoted by its parent i.e., p is an ancestor of c and has level one less than that of c.

Leaf A box that has no children.

Far-field That region wherein both multipole and Taylor series expansions are valid.

Near-field The {entire region} − {far-field}. This includes regions where only one, or neither, of the multipole expansion or Taylor series approximations are valid.

Colleague For the box of interest, b, a colleague is any of the same level near-field boxes, but not including b itself.

Consider a field box, b_f, centered at R_f and of side length Δs_f, and a source box, b_s, centered at R_s and of size Δs_s. Denote by $\|\cdot\|_\infty$ the distance in l_∞ space defined as,

$$\|\mathbf{R}\|_\infty \equiv \max_k \{|R_k|, k = 1,3\}, \qquad (11)$$

where **R** is any Cartesian vector with components, R_k.

A multipole expansion may be applied if the separation between boxes exceeds a certain multiple, M_G, of the *source* box size. This requirement can be stated in terms of a strict inequality constraint:

$$\|R_f - R_s\|_\infty > M_G\,\Delta s_s + \tfrac{1}{2}\max\{\Delta s_f, \Delta s_s\}. \quad (12)$$

Similarly, a Taylor series expansion is valid when the box separation exceeds a multiple, M_T, of the *field* box size. The resulting inequality is,

$$\|R_f - R_s\|_\infty > M_T\,\Delta s_f + \tfrac{1}{2}\max\{\Delta s_f, \Delta s_s\}. \quad (13)$$

In the present implementation, it is assumed that $M_G = M_T = M_S$, which not only simplifies the rules required to determined the types of interactions, but also assures symmetrical interaction between group pairs. M_S is the specified box separation parameter that is used to distinguish near-/far-field regions.

Denoting,

$$D_{sf} \equiv \|R_f - R_s\|_\infty - \tfrac{1}{2}\max\{\Delta s_f, \Delta s_s\}, \quad (14)$$

then between any pair of boxes, five types of interactions are considered.[83] These are:

A. Neither multipole expansion nor Taylor series approximations can be applied when determining the effect of b_s upon b_f, i.e.,

$$D_{sf} < M_S\min\{\Delta s_s, \Delta s_f\}. \quad (15)$$

Source box, b_s, is then said to be *adjacent* to box, b_f, and interactions between particles in the two boxes must be computed directly.

B. A Taylor series approximation is valid, but a multipole expansion is not, i.e.,

$$M_S\,\Delta s_f < D_{sf} < M_S\,\Delta s_s. \quad (16)$$

C. A multipole expansion is valid, but a Taylor series expansion is not, i.e.,

$$M_S\,\Delta s_s < D_{sf} < M_S\,\Delta s_f, \quad (17)$$

which is the dual of case *B*: if condition *C* applies, then so will condition *B* when the roles of boxes b_f and b_s are interchanged, with b_f acting as the source box and b_s as the field box.

D. Both the multipole and Taylor series expansions are applicable. This requires,

$$D_{sf} > M_S\max\{\Delta s_s, \Delta s_f\}. \quad (18)$$

E. No interaction occurs between the pair since the mutually induced effects have already been accounted for during interaction of their parent boxes.

Figure 3 shows the five possible interactions (A–E) of a field box, *F*, with all other boxes.

The set of source boxes, b_s, is determined by first identifying the coarsest level ancestor of b_f and this ancestor's colleagues. A tree search over the descendants of these coarsest level boxes is then conducted and the interactions, A → E, identified. Finally, the interactions are realized by effecting the appropriate multipole and/or Taylor series expansions.

The procedures used to determine interactions, A → E, involve only integer and comparison operations. Significant pruning occurs since, for example, case B need only be considered when b_f is a leaf, case C when b_s is a leaf, and case A when both b_s and b_f are leaves. Numerical experience shows that the CPU time involved in constructing the necessary pointer arrays is a small fraction (typically 1% or less) of the overall computation time. Note that a popular practice in implementing nested hierarchical calculations of this nature is to invoke recursion routines available for example in the C programming language. While this arguably produces more elegant and compact source code, its implementation is expensive in practice since

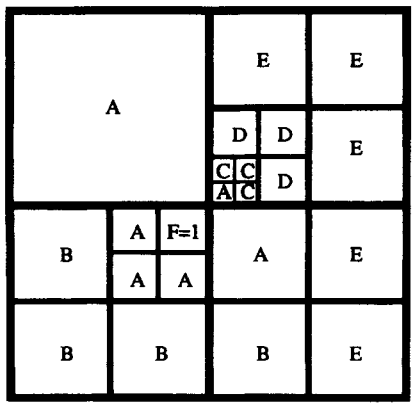

FIGURE 3. A two-dimensional example illustrating the five possible types of interactions with a field box, *F*. Only terminal boxes are shown and $M_S = 1$ is assumed. The letters correspond to the interactions described in the text.

each recursive call involves duplication of the calling routine and its data arrays. The present implementation does not employ such techniques and thus avoids duplicity.

Results and Discussion

The fast adaptive multipole method described above is employed to compute the electrostatic energy of a DNA circle and to examine the performance of a fast Monte Carlo algorithm in a one-step simulation of polyelectrolyte DNA at two different salt concentrations (specifically at the extremes of no added salt and the high salt regime). These results are presented in the following subsections.

COMPUTATION OF ELECTROSTATIC ENERGY OF CIRCULAR DNA

To examine the accuracy and speed enhancement of the fast adaptive multipole method, simple test calculations of the electrostatic energy of a DNA circle under the condition of no added salt (c_s = 0 M, where c_s is the concentration of NaCl) and high salt concentration (c_s = 1 M) were carried out. The results below were obtained using the direct and fast energy codes implemented on a Silicon Graphics 4D/440 computer system operating in the Center for Computational Chemistry at Rutgers University.

In the fast calculation, up to six levels in the hierarchy of the boxes are allowed and the maximum number of particles per box (N_c) is set to 20. Ideally, the value of N_c should be adjusted to minimize the CPU time. Setting N_c too large results in an excessive number of near-field interactions (in the extreme case of $N_c = N$, only direct interactions occur), whereas setting it too small leads to a large number of hierarchical boxes. The box separation parameter, M_S, which defines the near-field region, is taken as 1. Increasing M_S (i.e., $M_S \geq 2$) leads to lower error but at the cost of more direct interactions and consequently lower CPU efficiency. Optimum choice of M_S is a nontrivial matter and depends upon the number of terms retained in the expansions and the specified error bound desired by the user. The difficulty of selecting optimal parameter values, which are influenced by problem size and particle distribution, is an important one. Currently, they are selected on a trial and error basis guided by previous experience in operating this and other similar codes.[57] Development of more systematic procedures for adjusting the input parameters in order to achieve a user-specified error tolerance at minimum computation cost is anticipated in future work. At all levels, three terms in the multipole expansion and two terms in the Taylor series expansion are retained. Limiting the Taylor series to only two terms is found to incur minimal computational cost and generally to produce accurate results.

In all cases, the electrostatic energy is first computed with the direct method given by eq. (3). This is considered the "exact" solution and is used as a reference against which the approximate energies obtained with the fast method can be compared. The relative electrostatic energy error given by $(U_a - U_e)/U_e$, where U_a is the approximate electrostatic energy obtained by the fast method and U_e is the exact energy computed using the direct approach, serves as a measure of the accuracy and reliability of the approximations embedded in the fast method.

Figures 4a and 4b show the electrostatic energy of a DNA circle as a function of the number of phosphate groups in the chain, N, under the conditions of no added salt and high salt, respectively. The computed electrostatic energy curves corresponding to the direct and fast methods are effectively indistinguishable. The absolute relative error (i.e., absolute value of the relative error values shown in the insets of Fig. 4a, b) is less than 0.6% in all cases. Further calculations performed for a 2500 base pair DNA in the monovalent salt range c_s = 0–1 M show that a worst absolute relative error of 1% at c_s = 0.001 M is incurred, which is considered acceptable. Accuracy can be improved by increasing M_S. For example, setting M_S = 2 reduces the worst absolute relative error to 0.06% at both salt concentrations and all chain lengths (up to 5000 base pairs or 10,000 phosphate groups). These results confirm that the fast adaptive multipole method constitutes a reliable approximation with error control for electrostatic energy prediction of DNA.

Figure 5 depicts the total CPU time (in seconds) required to compute the electrostatic energy of the DNA as a function of the number of charged phosphate groups under the conditions of no added salt and high salt. Both the expected quadratic time complexity of the direct computation and the predicted $O(N)$ performance of the fast calculation are clearly evident from the slopes of the log–log plots. For both salt conditions, dramatic improvement in the computational speed is established. The stable trends of total CPU time with N

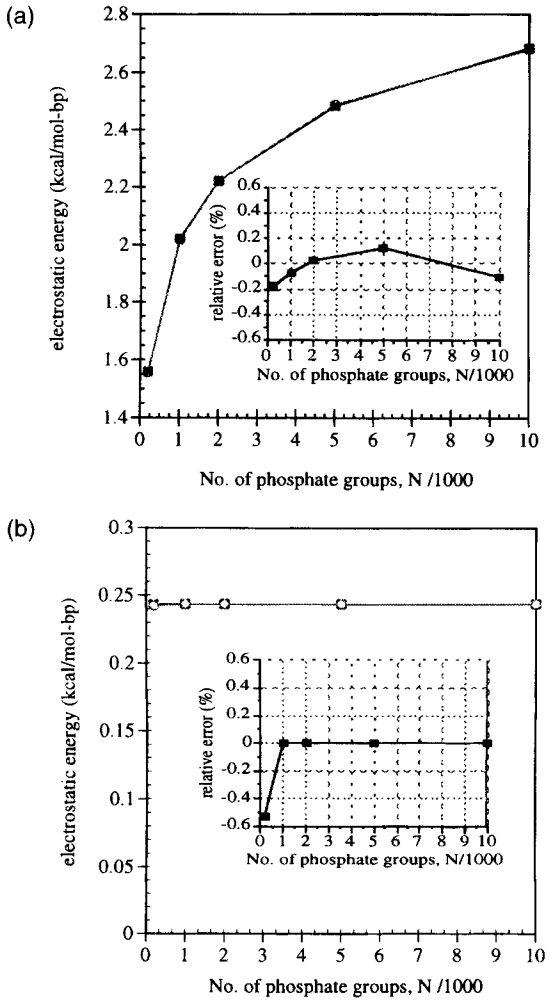

FIGURE 4. Comparison as a function of chain length (i.e., number of phosphate groups) of the electrostatic energy of a DNA circle computed using the direct and the fast method. The inset shows the relative error (as defined in the text) versus the number of phosphate groups of DNA. In (a) the salt concentration c_s is fixed at 0 M (no added salt) and in (b) c_s = 1 M.

in Figure 5 imply that the dramatic gains in performance are maintained for $N > 10^4$.

Figure 6 records the speedup factor of the fast method in the electrostatic energy computation versus the number of phosphate groups of DNA. The fast method, despite its additional overhead, is already faster than the direct approach for $N \geq 200$. Even for as few as 1000 charged phosphate groups, the adaptive algorithm is 10 times faster than the direct calculation. The most dramatic gains, however, are realized for large numbers of phosphate groups. With 10^4 phosphate groups (i.e., 5000 base pairs), remarkable speedup factors of

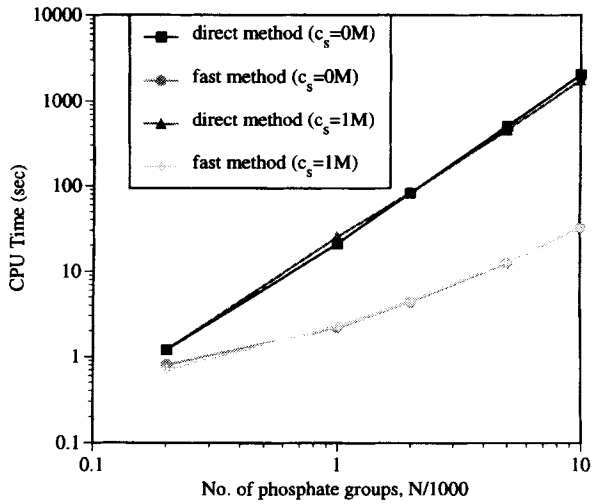

FIGURE 5. Comparison versus chain length of the CPU time (in seconds) for the calculation of the electrostatic energy at c_s = 0 M and 1 M using the direct and fast methods.

about 69 (c_s = 0 M) and 64 (c_s = 1 M), are obtained. Such order of magnitude reductions in CPU time render the simulation of large-scale, realistic DNA molecules feasible using current computing resources.

ONE STEP MONTE CARLO SIMULATIONS OF SUPERHELICAL DNA

Next, the accuracy and speed performance of the fast method incorporated into a Monte Carlo/

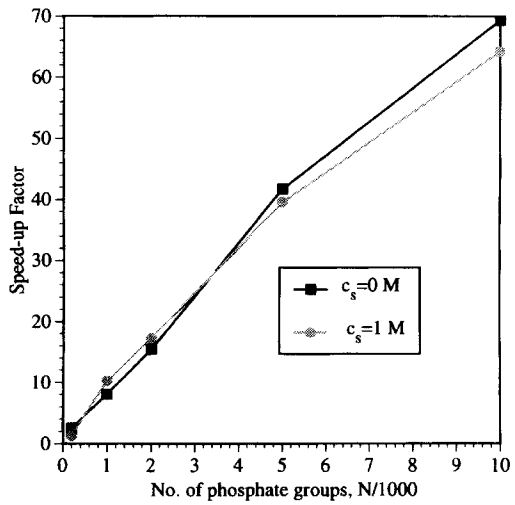

FIGURE 6. Plot of the speedup factor of the fast method versus the number of charged phosphate groups of DNA at c_s = 0 M and 1 M.

simulated annealing algorithm for polyelectrolyte DNA supercoils are demonstrated. In these simulations, the DNA duplex is treated as a thin, symmetric, linearly elastic charged rod. Forces opposing the deformation of the rod are partitioned locally into independent harmonic contributions from the changes of bending and twisting, described, respectively, in terms of the local curvature and twist of the DNA. Thus, the elastic energy term is composed of a bending and twisting contribution. Excluded volume effects are taken into account by means of a hard-sphere potential, which restricts the contacts of chain contour points to distances greater than some arbitrary limit (here set equal to 20 Å). As in the above calculations, the polyelectrolyte character of the charged phosphate groups in the chain backbone is accounted for by including the modified Debye–Hückel energy contribution given by eq. (3). The starting configuration in all the simulations is a perfect DNA circle represented by 14 *B*-spline controlling points. The linking number difference, a measure of the degree of supercoiling, is set equal to 3. The composite elastic/electrostatic energy is minimized under the constraints of ring closure and chain length. Full details of the combined elastic and electrostatic potential function are given elsewhere.[12]

In typical MC/simulated annealing studies of polyelectrolyte DNA, roughly $O(10^5)$ steps are involved. At each step, both the coordinates of the evenly spaced phosphorus atoms and the total energy of the DNA configuration are computed. In the first step of the MC calculation reported below, the phosphorus coordinates and total energy are computed twice, once for the initial configuration (i.e., the initial DNA circle) and again for the first perturbed DNA configuration. Complete details concerning the MC/simulated annealing algorithm are presented elsewhere.[9,12]

As above, the electrostatic energies computed using the fast method and direct summation are compared. The parameters in the calculation are identical to those used in the computations of the electrostatic energy discussed above. The performance speed is here defined as the total CPU time (in seconds) required for evaluating one step of the MC simulation.

As with the rigid DNA circles, excellent agreement between the electrostatic energies obtained with the direct and fast methods is observed with the absolute relative error in the modified Debye–Hückel energy being less than 0.5% in all cases. We have also performed long simulations (involving 50,000–500,000 configurational moves) of supercoiled DNA with up to 1500 base pairs (i.e., 3000 phosphate groups) using the direct and fast methods and have verified that the simulated annealing sequence converges to essentially the same minimum energy supercoiled configuration in both cases (our unpublished results). This further supports the use of low-order multipole/ Taylor series expansions with small near-field regions ($M_S = 1$) and provides additional evidence for the overall reliability and accuracy of the fast method in polyelectrolyte DNA simulation.

A comparison of the computation time (in seconds) taken in one step of Monte Carlo simulation versus the number of charged phosphate groups is presented in Figure 7 for the cases of no added salt and high salt. Again, the variation of total CPU time with the number of phosphate groups confirms the expected quadratic and $O(N)$ behaviors of the direct and fast methods, respectively. A computational speedup in excess of a factor of 50 is achieved for simulations of polyelectrolyte DNA with 10,000 phosphate groups (i.e., 5000 base pairs, a size typical of naturally occurring supercoiled plasmids[13]) when using the fast method compared to the direct approach. Furthermore, the robust trends in CPU times confirm that this performance improvement continues for realistic large-scale MC

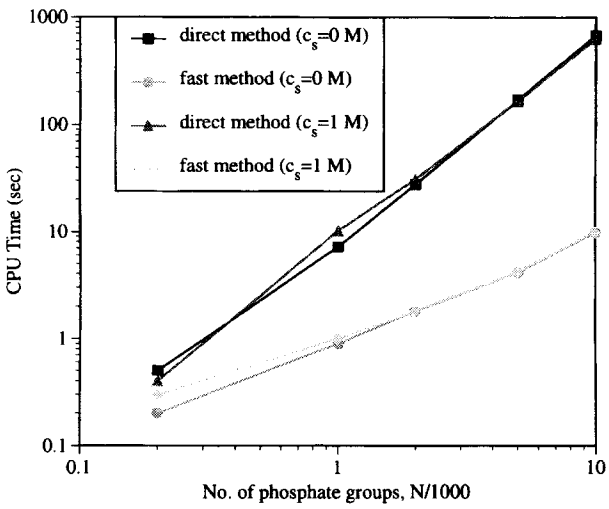

FIGURE 7. A plot of the CPU time (in seconds) required for evaluation of the first step of the Monte Carlo simulation of a supercoiled DNA polyelectrolyte for increasing chain lengths (i.e., phosphate groups) and under the conditions of no added salt (i.e., $c_s = 0$ M) and high salt concentration (i.e., $c_s = 1$ M) using the direct and fast methods.

DNA simulations involving $O(10^4-10^5)$ charged phosphate groups.

It should be pointed out that in all cases (i.e., at both salt concentrations and all chain lengths) the CPU time of one MC move is approximately three times that of the electrostatic energy computation of the rigid DNA circle presented in the last section. This increase reflects the additional operations required to determine the phosphorus coordinates and total energy (i.e., elastic and hard-sphere energy contributions in addition to the electrostatic energy term computed in the last section) for both the initial and the first perturbed DNA configuration.

Summary and Future Directions

A fast adaptive multipole algorithm for the computation of electrostatic interactions in DNA was successfully formulated and implemented on serial computers. By exploiting multipole and Taylor series expansions in conjunction with an adaptive hierarchical clustering scheme, the fast algorithm reduces the time complexity from $O(N^2)$ identified with direct summation of all pairwise interactions of phosphate groups to $O(N)$. The robustness, accuracy, and the anticipated asymptotic time complexity of the fast method over the direct calculation were verified in comparative studies of the electrostatic energy of rigid DNA circles and in supercoiled DNAs allowed to undergo configurational changes in one-step Monte Carlo simulations. Reductions in total CPU time by two orders of magnitude were achieved for DNA with $O(10^4)$ charged phosphate groups.

The above implementation of a fast multipole algorithm to the computation of electrostatic interactions in DNA is very encouraging since it enables conducting realistic numerical simulations of long supercoiled chains using existing computing resources (i.e., serial computers) with negligible loss in the numerical accuracy of the predicted modified Debye–Hückel energy. In fact, we have already been able to carry out Monte Carlo simulations of polyelectrolyte DNA supercoils with as many as 1500 base pairs (our unpublished data).

Future work is planned for enhancing the computational efficiency of the existing code. Modifications expected to lead to substantial improvements include the use of vectorization, storage minimization of the multipole and Taylor series coefficients, and optimization of adjustable code parameters, such as the number of particles permitted per box before subdivision occurs (N_c), box separation parameter M_S, and expansion orders of the multipole and Taylor series approximations.

The fast method is also amenable to parallel implementation for further CPU reduction. This option is currently being pursued and the fast scheme has been implemented on an Intel iPSC/860 parallel computer (our unpublished results). Careful attention to minimizing the communication overhead for large-scale computations and maximizing load balance so that the calculation is evenly distributed among the available processors[53] resulted in a parallel implementation that scales well with the number of processors. The parallel analysis was used to compute both electrostatic energy and forces and also the writhing number (i.e., a geometric quantity that measures the path of the global helix axis in space), which are all amenable to fast implementation. Preliminary results of an implementation of the fast algorithm on a 32 node machine for a planar ring containing 10^4 equally spaced particles with unit charge demonstrated an additional 23-fold reduction in CPU time compared to serial operation. This represents an $O(10^3)$ improvement compared to the same problem solved using direct summation on a scalar machine.[84]

Acknowledgments

Our research was supported by the NIH under SBIR Phase I Grant 1 R43 GM 50132-01. Calculations were performed at the Rutgers Center for Computational Chemistry. The authors thank Dr. Ramzi Kutteh (Batelle) for carefully reading the manuscript and for providing valuable comments. A glossary describing all the symbols used in the manuscript is available from the authors upon request (e-mail: fenley@chemf.rutgers.edu).

References

1. G. S. Manning, *Quart. Rev. Biophys.*, **179**, 181 (1978).
2. B. Jayaram, N. Aneja, E. Rajasekaran, V. Arora, A. Das, V. Ranganathan, and V. Gupta, *J. Sci. Ind. Res.*, **53**, 88 (1994).
3. A. T. Brünger, R. L. Campbell, G. M. Clore, A. M. Gronenborn, M. Karplus, G. A. Petsko, and M. M. Teeter, *Science*, **235**, 1049 (1987).
4. A. T. Brünger, *Ann. Rev. Phys. Chem.*, **42**, 197 (1991).
5. I. D. Kuntz, E. C. Meng, and B. K. Shoichet, *Acc. Chem. Res.*, **27**, 117 (1994).

6. D. L. Beveridge, S. Swaminathan, G. Ravishanker, J. M. Withka, J. Srinivasan, C. Prevost, S. Louise-May, D. R. Langley, F. M. DiCapua, and P. H. Bolton, In *Topics in Molecular and Structural Biology: Water and Biological Macromolecules*, Vol. 17, E. Westhof, W. Fuller, and S. Neidle, Eds., Macmillan, London, 1991, p. 165.
7. B. Jayaram and D. L. Beveridge, *J. Phys. Chem.*, **95**, 2506 (1991).
8. A. E. Love, *Treatise on the Mathematical Theory of Elasticity*, 4th ed., Dover, New York, Chaps. 18, 19, 1944.
9. M.-H. Hao and W. K. Olson, *Macromolecules*, **22**, 3292 (1989).
10. T. Schlick and W. K. Olson, *J. Mol. Biol.*, **223**, 1089 (1992).
11. A. V. Vologodskii and N. R. Cozzarelli, *Ann. Rev. Biophys. Biomol. Struct.*, **23**, 609 (1994).
12. M. O. Fenley, W. K. Olson, I. Tobias, and G. S. Manning, *Biophys. Chem.*, **50**, 255 (1994).
13. J. C. Wang, *J. Mol. Biol.*, **43**, 25 (1969).
14. C. F. Anderson and W. Bauer, *Biochemistry*, **17**, 594 (1978).
15. G. W. Brady, D. B. Fein, J. Lambertson, V. Grassian, D. Foos, and C. J. Benham, *Proc. Natl. Acad. Sci. USA*, **80**, 741 (1983).
16. T. C. Boles, J. H. White, and N. R. Cozzarelli, *J. Mol. Biol.*, **213**, 931 (1990).
17. M. Adrian, B. ten Heggeler-Bordier, W. Wahli, A. Z. Stasiak, A. Stasiak, and J. Dubochet, *EMBO J.*, **9**, 4551 (1990).
18. V. V. Rybenkov, N. R. Cozzarelli, and A. V. Vologodskii, *Proc. Natl. Acad. Sci. USA*, **90**, 5307 (1993).
19. S. Y. Shaw and J. C. Wang, *Science*, **260**, 533 (1993).
20. J. Bednar, P. Furrer, A. Stasiak, J. Dubochet, E. H. Egelman, and A. D. Bates, *J. Mol. Biol.*, **235**, 825 (1994).
21. C. L. Brooks III, B. M. Pettitt, and M. Karplus, *J. Chem. Phys.*, **83**, 5897 (1985).
22. R. J. Loncharich and B. R. Brooks, *Proteins*, **6**, 32 (1989).
23. H. Schreiber and O. Steinhauser, *Chem. Phys.*, **168**, 75 (1992).
24. J. Guenot and P. Kollman, *J. Comput. Chem.*, **14**, 295 (1993).
25. P. J. Steinbach and B. R. Brooks, *J. Comput. Chem.*, **15**, 667 (1994).
26. M. Saito, *J. Chem. Phys.*, **101**, 4055 (1994).
27. L. Greengard and V. Rokhlin, *Chem. Scripta*, **29A**, 139 (1989).
28. R. H. Stote, D. J. States, and M. Karplus, *J. Chim. Phys.*, **88**, 2419 (1991).
29. F. S. Lee and A. Warshel, *J. Chem. Phys.*, **97**, 3100 (1992).
30. M. Saito, *Mol. Simul.*, **8**, 321 (1992).
31. J. A. Board Jr., J. W. Causey, J. F. Leathrum Jr., A. Windemuth, and K. Shulten, *Chem. Phys. Lett.*, **198**, 89 (1992).
32. J. Shimada, H. Kaneko, and T. Takada, *J. Comput. Chem.*, **14**, 867 (1993).
33. J. N. Glosli and M. R. Philpott, *Technical Report 93-03*, IBM, San Jose, CA, 1993.
34. V. Dillet, D. Rinaldi, J. Angyan, and J. Rivail, *Chem. Phys. Lett.*, **202**, 18 (1993).
35. R. Kutteh and J. B. Nicholas, *Comp. Phys. Comm.*, to appear.
36. T. Darden, D. York, and L. Pedersen, *J. Chem. Phys.*, **98**, 10089 (1993).
37. D. York, A. Wlodawer, L. Pederson, and T. Darden, *Proc. Natl. Acad. Sci. USA*, **91**, 8715 (1994).
38. D. York and W. Yang, *J. Chem. Phys.*, **101**, 3298 (1994).
39. A. M. Mathiowet, A. Jain, N. Karasawa, and W. A. Goddard III, *Proteins*, **20**, 227 (1994).
40. C. G. Lambert, *Technical Report 94-004*, Duke University, Durham, NC, 1994.
41. W. D. Elliot and J. A. Board Jr., *Technical Report 94-005* Duke University, Durham, NC, 1994.
42. J. A Board Jr., Z. S. Hakura, W. D. Elliott, D. C. Gray, W. J. Blanke, and J. F. Leathrum Jr., *Technical Report 94-002*, Duke University, Durham, NC, 1994.
43. J. Shimada, H. Kaneko, and T. Takada, *J. Comput. Chem.*, **15**, 28 (1994).
44. C. Niedermeier and P. Tavan, *J. Chem. Phys.*, **101**, 734 (1994).
45. F. Figueirido, G. S. Del Buono, and R. M. Levy, *J. Chem. Phys.*, **103**, 6133 (1995).
46. V. Rokhlin, *J. Comp. Phys.*, **60**, 187 (1985).
47. L. Greengard and V. Rokhlin, *J. Comp. Phys.*, **73**, 325 (1987).
48. K. Nabors, F. T. Korsmeyer, F. T. Leighton, and J. White, *SIAM J. Sci. Comput.*, **15**, 713 (1994).
49. A. Leonard, *J. Comp. Phys.*, **37**, 289 (1980).
50. A. Leonard, *Ann. Rev. Fluid Mech.*, **17**, 523 (1985).
51. K. Chua, A. Leonard, F. Pepin, and G. Winckelmans, *Proc. ASME*, Chicago, IL, November 1988.
52. L. Van Dommelen and E. A. Rundensteiner, *J. Comp. Phys.*, **83**, 126 (1989).
53. K. Chua and T. R. Quackenbush, *Parallel Computational Fluid Dynamics*, Elsevier Science Publishers, Amsterdam, 1992.
54. K. Chua and T. R. Quackenbush, *AIAA Paper 92-2624, Proceedings of the 10th AIAA Applied Aerodynamics Conference*, Palo Alto, CA, June 1992.
55. C. I. Draghicescu and M. Draghicescu, *J. Comp. Phys.*, to appear.
56. A. Ghoniem, *Numerical Approaches to Combustion Modeling*, AIAA, Washington, DC, 1991, p. 305.
57. K. Chua, T. R. Quackenbush, and A. Leonard, *Continuum Dynamics, Inc. Report No. 92-03*, prepared under SBIR Phase I Grant No. DE-FG05-91ER81207 for the U.S. Department of Energy, June 1992.
58. K. Chua, A. H. Boschitsch, P. D. Koumoutsakos, G. S. Winckelmans, and A. Leonard, *AIAA Paper No. 94-0675*, 32nd Aerospace Sciences Meeting & Exhibit, Reno, NV, January 1994.
59. H.-Q. Ding, N. Karasawa, and W. A. Goddard III, *J. Chem. Phys.*, **97**, 4309 (1992).
60. H.-Q. Ding, N. Karasawa, and W. A. Goddard III, *Chem. Phys. Lett.*, **196**, 6 (1992).
61. S. R. Lustig, J. J. Cristy, and D. A. Pensak, *Mater. Res. Soc. Symp. Proc.*, **278**, 9 (1992).
62. W. Li, R. K. Kalia, S. De Leeuw, A. Nakano, D. Greenwell, and P. Vashishta, *Mater. Res. Soc. Symp. Proc.*, **291**, 267 (1993).
63. S. J. Aarseth, J. R. Gott III, and E. L. Turner, *Astrophys. J.*, **228**, 664 (1979).
64. A. W. Appel, *SIAM J. Sci. Stat. Comput.*, **6**, 85 (1985).
65. J. Barnes and P. Hut, *Nature*, **324**, 446 (1986).
66. G. C. Fox, P. Hipes, and J. Salmon, *Proceedings of Supercomputing*, Reno, NV, 1989.
67. R. W. Hockney, S. P. Goel, and J. W. Eastwood, *J. Comp. Phys.*, **14**, 148 (1974).
68. S. Pfalzner and P. Gibbon, *Comp. Phys. Commun.*, **79**, 24 (1994).

69. L. Greengard, *Comp. Phys.*, **4**, 142 (1990).
70. L. Greengard, *Science*, **265**, 909 (1994).
71. C. K. Birdsall and D. Fuss, *J. Comp. Phys.*, **3**, 494 (1969).
72. J. P. Christiansen, *J. Comp. Phys.*, **13**, 363 (1973).
73. R. Hockney and J. Eastwood, *Computer Simulation Using Particles*, McGraw–Hill, New York, 1981.
74. P. R. Spalart and A. Leonard, *AIAA-81-1246*, American Institute of Aeronautics and Astronautics (AIAA) 14th Fluid and Plasma Dynamics Conference, Palo Alto, CA, 1981.
75. M. E. Tuckerman, B. J. Berne, and A. Rossi, *J. Chem. Phys.*, **94**, 1465 (1991).
76. C. R. Anderson, *SIAM J. Sci. Stat. Comput.*, **13**, 923 (1992).
77. L. Greengard, *The Rapid Evolution of Potential Fields in Particle Systems*, MIT Press, Cambridge, MA, 1988.
78. J. Carrier, L. Greengard, and V. Rokhlin, *SIAM J. Sci. Stat. Comput.* **9**, 669 (1988).
79. C. A. White and M. Head–Gordon, *J. Chem. Phys.*, **101**, 6593 (1994).
80. M. E. Mortenson, *Geometric Modeling*, Wiley, New York, 1985.
81. M.-H. Hao and W. K. Olson, *Biopolymers*, **28**, 873 (1989).
82. R. Chandrasekaran and S. Arnott, *Landolt–Börnstein Numerical Data and Functional Relationships in Science and Technology, New Series, Group VII: Biophysics, Volume 1, Nucleic Acids, Subvolume b, Crystallographic and Structural Data*, W. Saenger, Ed., Springer–Verlag, Berlin, 1989, p. 31.
83. J. H. Strickland and D. E. Amos, *AIAA J.*, **30**, 737 (1992).
84. K. Chua, A. H. Boschtisch, A. J. Bilanin, W. K. Olson, and M. O. Fenley, *Continuum Dynamics, Inc. Report No. 94-02*, prepared under SBIR Phase I Grant No. 1 R43 GM50132-01 for the Department of Health and Human Services, National Institutes of Health, Princeton, May 1994.